(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
20 December 2001 (20.12.2001)

PCT

(10) International Publication Number
WO 01/97070 A1

(51) International Patent Classification[7]: G06F 17/00

(21) International Application Number: PCT/US01/18784

(22) International Filing Date: 12 June 2001 (12.06.2001)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
09/593,150 14 June 2000 (14.06.2000) US

(71) Applicant: **ARTESIA TECHNOLOGIES, INC.** [US/US]; Suite 400, 700 King Farm Boulevard, Rockville, MD 20850 (US).

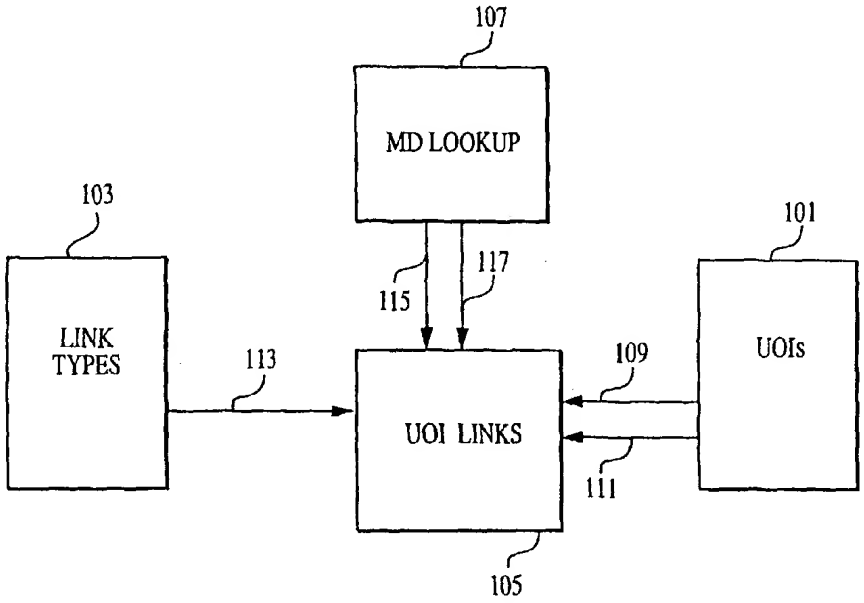
(72) Inventors: **CHAKRAVARTY, Dipto**; 700 King Farm Boulevard, Suite 400, Rockville, MD 20850 (US). **MC-GONIGLE, Lynn**; 700 King Farm Boulevard, Suite 400, Rockville, MD 20850 (US). **TAM, Jack**; 700 King Farm Boulevard, Suite 400, Rockville, MD 20850 (US). **JENKINS, Edward**; 700 King Farm Boulevard, Suite 400, Rockville, MD 20850 (US). **NOROUZI, Maryam**; 700 King Farm Boulevard, Suite 400, Rockville, MD 20850 (US). **MEDLIN, Roger**; 700 King Farm Boulevard, Suite 400, Rockville, MD 20850 (US). **HARPER, Ben, Dale**; 700 King Farm Boulevard, Suite 400, Rockville, MD 20850 (US). **BOLLINGER, Gary**; 700 King Farm Boulevard, Suite 400, Rockville, MD 20850 (US). **FALL, Malick**; 700 King Farm Boulevard, Suite 400, Rockville, MD 20850 (US).

(74) Agents: **NICHOLSON, Cynthia, K.** et al.; Hale and Dorr LLP, 1455 Pennsylvania Avenue, N.W., Washington, DC 20004 (US).

(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ,

[Continued on next page]

(54) Title: METHOD AND SYSTEM FOR LINK MANAGEMENT



(57) Abstract: A method and system for digital asset link management. The system includes information on units of information UOI (101), information on link types (103) and storage relating to the UOI links (105). Optionally included is an ancillary metadata lookup table (107). A source value (109) and a destination value (111) are provided from the UOIs (101) to the UOI links (105). The link type (113) is obtained from the link types file (103) and provided to the UOI links (105). Optionally, the source type (115) and destination type (117) are obtained from the metadata lookup table (107).

WO 01/97070 A1

NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, UZ, VN, YU, ZA, ZW.

(84) **Designated States** *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published:
— *with international search report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

Method and System for Link Management

BACKGROUND OF THE INVENTION

RELATED APPLICATIONS

This application claims priority from U.S. Patent Application serial number 09/593,150, filed June 14, 2000, incorporated herein by reference.

FIELD OF THE INVENTION

The present invention consists of a method and system for managing links between digital assets. More particularly, it concerns a method and system for managing the relationships, that is, the links themselves, between digital assets of any type.

2. DESCRIPTION OF THE RELATED ART

Companies devote much time and many resources to the creation, storage and retrieval of digital assets. In this patent, the term "digital assets" is intended to include assets of any type which have been digitized. Such assets could be, for example, logos, trademarks, marketing materials, press releases, newspapers, publications of all types, print advertisements, internet advertising, broadcast advertising, and so forth. Digital assets are therefore anything which can be digitized and which has value to a company in a stored electronic form.

Even a midsize company would probably require a large database management system in order to track, a typical number of objects. The problem increases exponentially with the number of objects in a database, and is therefore a much more serious issue for a company with a large database.

Many of these objects are related. For example, consider an advertisement that includes several piece of animation, music and a logo. Each of these is a separate digital asset; each is related. In a database management system, containing a very large number of assets, it can be extremely difficult to manage not only the digital assets but also the multiple relationships between assets.

If the company does store its digital assets, it is difficult to track and evaluate the currency of a particular digital asset. In other words, tracking whether a particular digital asset is still a valid version is difficult. In the past, digital assets and their relation to other digital assets, have been tracked manually if at all. Some semi-automated systems are known, in which an application may merely access one view of an asset.

Certain versions of digital asset management have been attempted for very limited applications. For example, Vignette Corp. offers a product under the trademark IRM. This product manages relationships between web pages on a web site. Unfortunately, this product does not take into consideration objects of undefined depth. Moreover, it is geared to a particular application.

Many conventionally available systems are content dependent. For example, the QUARK™ desktop publishing software is content dependant. These types of implementations are simply not

medium agnostic. Unfortunately, in the situation where the digital assets may include, streaming media, audio, video and/or text, the underlying content of the digital assets should not be relevant.

Other systems have attempted to track relationships between elements using a table. Consider, for example, U.S. Patent No.: 5,832,495, Gustman, in which relationships are tracked between elements by using a table. Unfortunately, the table quickly becomes very dense in relation to the number of links. As the number of elements increases, the size of the table becomes almost unmanageable.

There thus remains a need for a system and method which can track relationships between digital assets, where the content of the digital asset itself is irrelevant.

BRIEF SUMMARY OF THE INVENTION

It is therefore an object of the present invention to provide a method and system for tracking digital assets in a manner which can handle a large number of digital assets, and in a way in which the content and the asset is irrelevant.

According to the invention, there is provided a method and system for digital asset link management. Electronic storage is accessed, the electronic storage containing at least one link relation, wherein each link relation includes a source identifier, a destination identifier, and a link type. It is determined, from the source identifier and destination identifier, a corresponding source digital asset and a corresponding destination digital asset. The process of determining includes determining the relation type between the source digital asset and destination digital asset. There are provided many link relations, source digital assets, and destination digital assets. The source digital assets included in at least a portion of the link relations are destination digital assets in another portion of the link relations.

Optionally, the source identifier and destination identifier each further has a metadata type. It is determined, for the at least one link relation, one or more attributes corresponding to each metadata type. Optionally, the source digital asset and destination digital asset each further includes a version indication.

An initial identifier is provided, and the system searches (via traversal or browser) for the electronic storage for any link relation having one of a source identifier and destination identifier corresponding to the initial identifier. The link relation may be traversed to a next linked link relation, wherein one of the source identifier and destination identifier of the link relation corresponds respectively to one of the destination identifier and source identifier of the next linked link relation.

Optionally, a bulk import is provided to create link relations, and the imported link relations are stored in the electronic storage.

These and other objects, features and advantages of the present invention are readily apparent from the following drawings and detailed description.

BRIEF DESCRIPTION OF THE DRAWINGS

The invention is described in connection with the drawings:

FIG. 1 is a block diagram of the system for link management.

FIG. 2 illustrates a table of UOIs (units of information).

FIG. 3 is an exemplary table of link types.

FIG. 4 is an exemplary table of UOI links.

FIG. 5 is an exemplary Metadata lookup table.

FIG. 6 is an example state diagram illustrating a link life cycle.

FIG. 7 is a block diagram illustrating the tool for visualization of a digital asset using the present system.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

The present invention seeks to solve the problems outlined above by tracking the link relationship between digital assets, and giving the source and destination for each link much less relevance. Reference is made to Figure 1. The link asset management system includes information on UOIs 101, information on link types 103, and storage relating to the UOI links 105. Optionally included is an ancillary Metadata lookup 107.

A "UOI" is a unit of information. It refers to an atomic unit of a digital asset, i.e., the smallest unit of information that has value, for example, a logo, text, etc.

The UOIs 101 contains unique identifiers corresponding to digital assets, preferably stored as a table. The link types 103 is preferably stored as a table, in order to correlate link types to a description relation between assets. Examples of link types include "brother of _____", "container of _____". The UOI links 105 contain link relations, i.e., corresponding sources, targets and links (or link types) between each corresponding source and target digital asset.

The term "link" as used herein means a relation between two assets. A "link type" is intended to accommodate industry specific vocabulary; examples include: "a parent of _____", "a container of _____", "attorney of _____". A "relationship" is an independent type of link, URL/URN or a hot spot. Thus each "link type" is a user defined relationship.

The Metadata ("MD") lookup 107 is preferably provided, although it is not necessary. Every UOI in the UOIs 101 references Metadata assets (for example, file type, file format, other attributes, etc. about the data). The MD lookup 107 contains, by Metadata type, the expected attributes of each type of Metadata. These attributes include, for example, that a TIF bit file type has specific Metadata associated therewith.

As is illustrated in Figure 1, a source value 109 and a destination value 111 are provided from the UOIs 101 to the UOI links 105. Also, the link type 113 is obtained from the link types file 103 and provided to the UOI links 105. Optionally, the source type and destination type 115, 117 are obtained from the Metadata lookup table 107.

As is illustrated in Figure 2, the UOIs include information for each UOI on UOI identification 201, and logical UOI identification 203. The UOIs also preferably includes version 205 and an

indication of whether this is the latest version 207. The UOI identification is a unique identifier corresponding to each unit of information. The logical UOI identification 203 is a logical identifier corresponding to the unit of information. In the example given in Figure 4, and referring to Figure 2 and 3 the application program will determine that UOIb is a destination of UOIc and a source of UOIa. UOIa is determined to have the logical UOI identification of "Diagram", UOIb identification the logical UOI of "Chapter 1 ", and UOIc identification has a logical UOI identification of "Book". The link between UOIa and UOIb is "TYPEa", i.e., "is container of". Thus, "Book" is he container of "Chapter 1". The foregoing is given by way of example, of course, without limitation to specific link types and logical UOI identifications. Thus, the table minimally contains each unique UOI identification and each corresponding logical UOI identification, in order that the relatively short UOI identification can be stored and the logical UOI identification can be much longer.

Also included in the UOIs 101, preferably, is an indication of the version 205 of each UOI. In this manner, multiple versions can be simultaneously stored. Also, preferably there is an indication of whether that version is in fact the latest version 207.

Thus, by searching the UOIs table by UOI identification, one can determine the corresponding logical UOI identification.

Reference is now made to Figure 3, illustrating the link types 103. The link types includes each of the link type codes 301, and the corresponding link type names 303. Preferably, the link types 103 is in the form of a table. The link type code can be any appropriate code. The link type name preferably takes the format of "is _____ of". The link type names are preferably determined by the user. By reference to the link types 103, one can determine the type of link between the two values.

Optionally included information with each of the link types is an indicator of whether the latest version is to be used, a reciprocal link type, any particular behavior, a status, and a description. The version connotes revisions of link participants. The reciprocal link type can best be understood by example. If, for example, the link type is "a parent of", the reciprocal link type is "a child of." The behavior is the semantic meaning associated with the specific link types. The status is a boolean description such as active versus inactive. The description describes the relationship between the UOI-s.

Figure 4 illustrates the UOI links 105. The UOI links includes a source UOI 401, a destination UOI 403, and a link type 405. Preferably, the source UOI and the destination UOI are stored as UOI identifications, and the link types are stored as link type codes, in a table. The table supports multiple source UOIs corresponding to a single destination UOI and conversely multiple destination UOIs corresponding to a single source 205. In the UOI link table, preferably, each link is stored as a source identification and a destination identification with the link type. The schema is therefore generic in that it will support linking of two digital assets. Optional information stored in the UOIs also includes source type, destination type, sequence number, a flag to indicate use of the latest version, status, and status date.

An application program can access the UOI links based on either a source UOI or a destination UOI. Based on the source (or destination) point, the application's program may obtain all of the corresponding destinations (or sources) and determine the corresponding link types. Next, by reference to the UOIs the source and destination logical UOI identifications are obtained. Also, by accessing the link types, the link type name is determined. Given a particular source or destination, the application table can then traverse to each next source or destination UOI and/or logical UOI, and can indicate to the user the link type name.

In the example given in Figure 4, and referring to Figure 2 and 3 the application program will determine that UOIb is a destination of UOIc and a source of UOIa. UOIa is determined to have the logical UOI identification of "body.htm", UOIb identification the logical UOI of "charger.bmp", and UOIc identification has a logical UOI identification of "main.htm". The link between UOIa and UOIb is "TYPEa", i.e., "is container of". Thus, body.htm is the container of "charger.bmp". The foregoing is given by way of example, of course, without limitation to specific link types and logical UOI identifications.

Figure 5 illustrates the Metadata lookup. The Metadata lookup 107 includes a Metadata type 501 and corresponding Metadata attributes 503 for each Metadata type. Each Metadata type may have associated with it certain attributes. The attributes would be dependent upon the type of the Metadata. For example, if the Metadata type is a code for a GIF file, the attributes would be specific to GIF files. Typically, the attributes would include the name of the Metadata type (such as a file type), and category. Category is a taxonomy or classification hierarchy.

The Metadata lookup is advantageously implemented as a table and is optionally included. It enhances the efficiency of the system to be able to reference a specific Metadata type and rapidly determine the expected attributes. Nevertheless, although the Metadata lookup table is preferred, it is ancillary and may be omitted.

Figure 6 is a state table illustrating a life cycle of a link. At step 601, a link is created. Following creation, the link may be traversed 603 by the application program. At least two types of traverse may be provided: searching 613 and browsing 611. The search traverse 613 permits a user to search for a specific source, destination or link type. The browse function 611 permits a user, whose application is presently located at a particular link, source or destination to traverse the link forward or backward to linked sources and destinations. A link may also be updated 605. A link may be deleted 607. Advantageously, deletion does not permanently remove the link. Thus, a link may be undeleted at 609. In order to permanently delete a link, it is preferable to purge 615 the link. Providing a separate process for delete and purge permits deletions to occur and then be undeleted if in fact the deletions were not desired.

In order to create a link, the link type should be defined by name, for example "is parent of", or "is payment of". Permitting a user to name the link types allows the creation of arbitrary relationships based on user defined terminology. If reciprocal relations are supported by the system, link creation is

an appropriate time to encourage the user to define the reciprocal relation, for example, "is child of" is the reciprocal relation to "is a parent of". The user could add additional link types to the ongoing database.

Once link types are created, a source and destination asset can be created. If the link types are created, they are added to the link type table 103. As source and destination assets are created, in connection with a particular relationship, the logical UOI identification, UOI identifications are added to the UOI; and the source UOI and the corresponding destination UOI and corresponding link types are added to the UOI link type table 105.

Links may be traversed as follows: Reference is made to both Figure 6 and Figure 7. Figure 7 illustrates a sliding window 703 and an exemplary portion of linked assets. Here, the linked assets include the current UOIs 701, the source UOIs at 705a, 705b, and the destination UOI 707a, 707b and 707c. In the present example, the brochure 705a is a parent of the press release 701; and the web page 705b is also the container of the press release 701. The press release itself is container of logo 707a, text 707b. The press release 701 is also "source of" ticker 707c.

The sliding window 703 is currently located on a press release 701. The sliding window can be moved to anyone of the linked source objects or destination objects corresponding to the press release. When located on one of the source objects or destination objects, the sliding window would indicate for that UOI, all of its source and destination links.

Reference is made back to Figure 6. The capability to update the UOIs is preferably provided. Update would permit the system to modify the relationships between UOIs. For example, an existing link type between a source and a destination UOI could be replaced with a different link type. Alternatively, a source or destination UOI could be replaced with a changed UOI.

The delete function illustrated in Figure 6, provides the capability to unlink the asset from the relationship. Note that the delete function does not delete the asset.

The digital asset management life cycle is as follows. In order to create UOIs, a system preferably has a preexisting list of the assets, and assigns to each of the assets a unique identification. Alternatively, assets may be created and added on the fly, and unique identifications added on the fly. As a first step, thus, UOIs are ingested, imported or loaded. As a second step, any ancillary Metadata is preferably created and the Metadata table is loaded. As a third step, each of the digital assets are assigned into the UOI table. As a forth, optional step, the assets are linked and placed into the UOI link table. As a fifth step, the currently linked assets may be searched and viewed, as discussed above in the link life cycle.

Assets may also be linked during a bulk import. To define a relation between any two assets, a link selection will be added within the UOIS tag off the XML input file, as show below:

DTD modification:

```
<!ELEMENT LINK EMPTY>
<!ATTLIST LINK      RELATION-TYPE CDATA #REQUIRED
                    RELATED-ASSET ENTITY          -- or CDATA, dependent
                                                  -- on implementation.
```

SGML/XML instance

```
<UOIS . . . >
    <LINK
        RELATION-TYPE="is-brother-of"
        RELATED-ASSET="C:\joshua.gif"
        SEQUENCE_NUM="1"
        USE_LATEST_VERSION="N"
    />
```

The following are descriptions of preferred tags within the LINK tag:

RELATION-TYPE: should match an existing NAME in the LINK_TYPES table

RELATED-ASSET: one way to uniquely identify the destination asset to which the current asset will be linked. If this tag is used, the value should be a fully qualified path to the current location of the file.

PO-REFERENCE: a second way to uniquely identify the destination asset to which the current asset will be linked. If this tag is used, the value should match the PO_REFERENCE value, in the UOIS section, of another asset in the same input file.

ASSET-ID: a third way to uniquely identify the destination asset to which the current asset will be linked. This tag can be used to link assets being imported to assets in the repository.

SEQUENCE-NUM: this optional tag may be used to provide a value, which will determine the position of the related assets among other destination assets of the same source asset and relation. If this tag is not used, no sequence is imposed on the destination assets.

USE-LATEST VERSION: this optional tag may be used to indicate whether a hard or soft link should be created to the destination asset. If this tag is not used, a hard link will be created by default. This is equivalent to the tag having a value of 'N'.

The following rules are advantageously used to guide the creation of relations between asset pairs during bulk import:

Links may be created between pairs of assets created during the same bulk import job. Linking new assets to existing repository assets should not be supported through bulk import.

Reciprocal relationships may be created automatically. If reciprocal relations are automatically created, they should be created for all relations for which a reciprocal link type has been specified in the LINK_TYPES table via link administration. If there is no reciprocal link type (e.g., value is NULL), no reciprocal relation is created. If reciprocal relations are *not* to be automatically created, this

functionality may still be provided by explicitly including the reciprocal LINK stanza in the import file. Reciprocal relations may then be defined on an as needed basis for each asset pair.

There should be no limitation on the number of relations that may be defined via bulk import.

There should be no limitation on the number of different link types that are used in defining relations via bulk import. That is, each asset pair may have a different link type from the others in the same file. In contrast, interactive import which preferably assumes the same link type for all asset pairs in the same import job.

The foreign key constraints drawn between the UOI_LINKS and the UOIs are optional as the source and destination assets need not be UOIs.

The functionality of a "hard" or "soft" link if desired, may be represented in the data model by the USE_LATEST_VERSION information in the UOIs. A soft link is recorded with a value of 'Y' in this column. The default value is 'N'. There is interdependency with check-in when a link's destination is UOI and the USE_LATEST_VERSION column is 'Y'. When the version number for the corresponding logical UOI_ID is incremented and a new UOI_ID assigned to the new version on the UOIS table, the logic should be implemented via a trigger. Given that, product assembly will be able to use the DEST_VALUE directly and not have to determine the newest version.

While the preferred mode and best mode for carrying out the invention have been described, those familiar with the art to which this invention relates will appreciate that various alternative designs and embodiments for practicing the invention are possible, and will fall within the scope of the following claims.

What is claimed is:

1. A method for digital asset link management, comprising the steps of

(a) accessing electronic storage containing at least one link relation, wherein each link relation includes a source identifier, a destination identifier, and a link type; and

(b) determining, from the source identifier and destination identifier, a corresponding source digital asset and a corresponding destination digital asset.

2. The method of claim 1, wherein the determining step fiirther includes determining the relation type between the source digital asset and destination digital asset.

3. The method of claim 1, wherein there are provided a plurality of link relations, a plurality of source digital assets, and a plurality of destination digital assets.

4. The method of claim 1, wherein the source digital assets included in at least a portion of the link relations are destination digital assets in another portion of the link relations.

5. The method of claim 1, wherein the source identifier and destination identifier each further has a metadata type, and further comprising the step of determining, for the at least one link relation, a plurality of attributes corresponding to each metadata type.

6. The method of claim 1, wherein the source digital asset and destination digital asset each further includes a version indication.

7. The method of claim 1, further comprising the step of providing an initial identifier and searching the electronic storage for any link relation having one of a source identifier and destination identifier corresponding to the initial identifier.

8. The method of claim 1, further comprising the step of creating a second link relation and storing said second link relation in said electronic storage.

9. The method of claim 1, further comprising the step of traversing at least one link relation to a next linked link relation, wherein one of the source identifier and destination identifier of the link relation corresponds respectively to one of the destination identifier and source identifier of the next linked link relation.

10. The method of claim 1, further comprising the step of performing a bulk import to create a plurality of link relations, and storing said imported link relations in said electronic storage.

11. A system for digital asset link management, comprising:

(a) electronic storage containing at least one link relation, wherein each link relation includes a source identifier, a destination identifier, and a link type; and

(b) a plurality of digital assets, including, for each source identifier and each destination identifier, a corresponding source digital asset and a corresponding destination digital asset.

12. The system of claim 11, wherein the link type corresponds to one of a plurality of relation types between the digital assets.

13. The system of claim 11, wherein there are provided a plurality of link relations, a plurality of source digital assets, and a plurality of destination digital assets.

14. The system of claim 11, wherein the source digital assets included in at least a portion of the link relations are destination digital assets in another portion of the link relations.

15. The system of claim 11, wherein the source identifier and destination identifier each further has a metadata type, each metadata type corresponding to a plurality of attributes.

16. The system of claim 11, wherein the source digital asset and destination digital asset each further includes a version indication.

17. The system of claim 11, further comprising a search routine, responsive to an initial identifier, for searching the electronic storage for any link relation having one of a source identifier and destination identifier corresponding to the initial identifier.

18. The system of claim 11, further comprising means for creating a second link relation and storing said second link relation in said electronic storage.

19. The system of claim 11, further comprising a browser, for traversing at least one link relation to a next linked link relation, wherein one of the source identifier and destination identifier of the link relation corresponds respectively to one of the destination identifier and source identifier of the next linked link relation.

20. The system of claim 11, further comprising a means for a bulk import to create a plurality of link relations, and to store said imported link relations in said electronic storage.

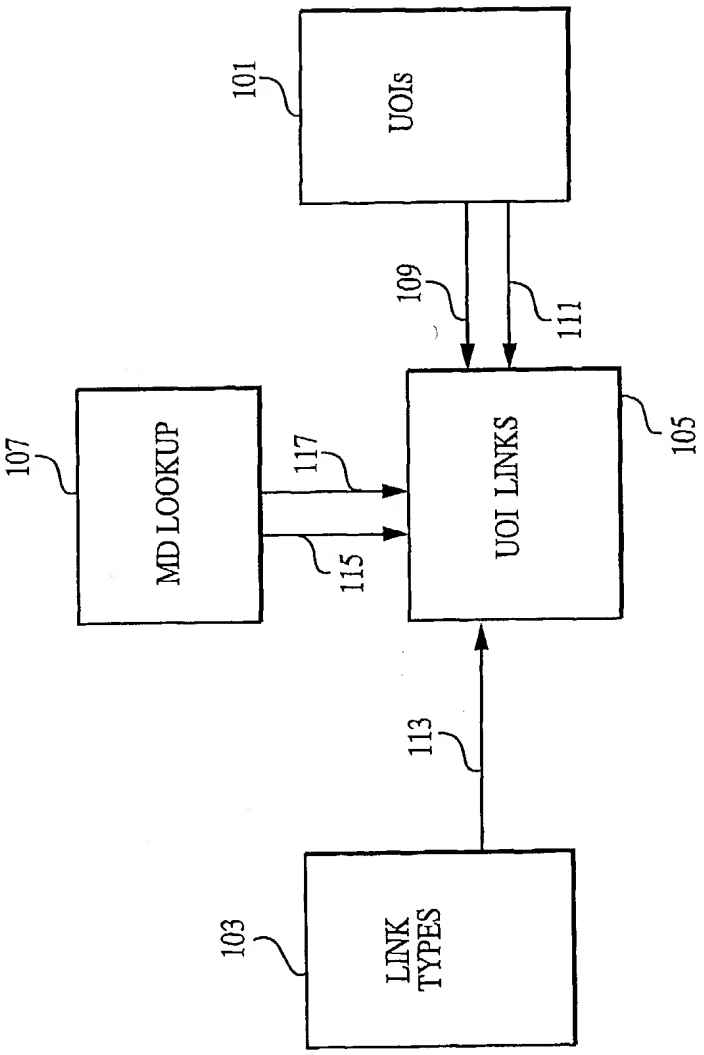

101
UOIs
109
111
107
MD LOOKUP
117
115
UOI LINKS
105
113
103
LINK TYPES

FIG. 1

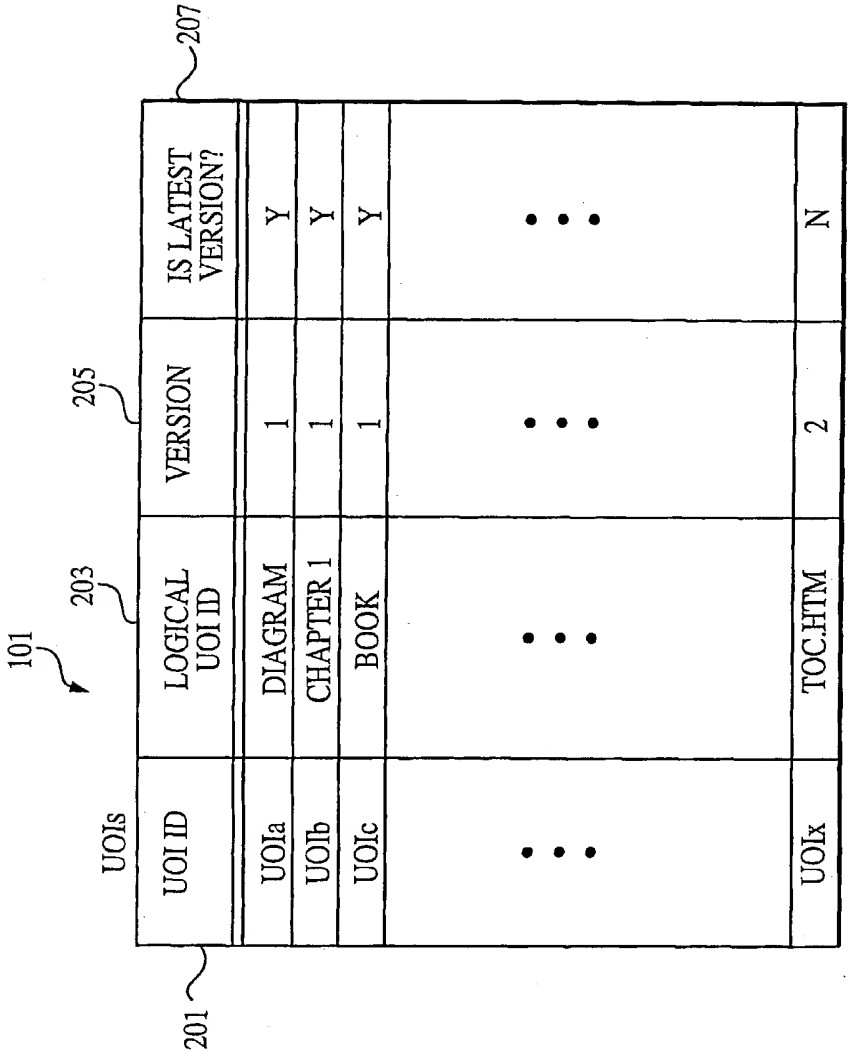

101
201
203
205
207
UOIs
UOI ID
LOGICAL UOI ID
VERSION
IS LATEST VERSION?
UOIa
DIAGRAM
1
Y
UOIb
CHAPTER 1
1
Y
UOIc
BOOK
1
Y
UOIx
TOC.HTM
2
N

FIG. 2

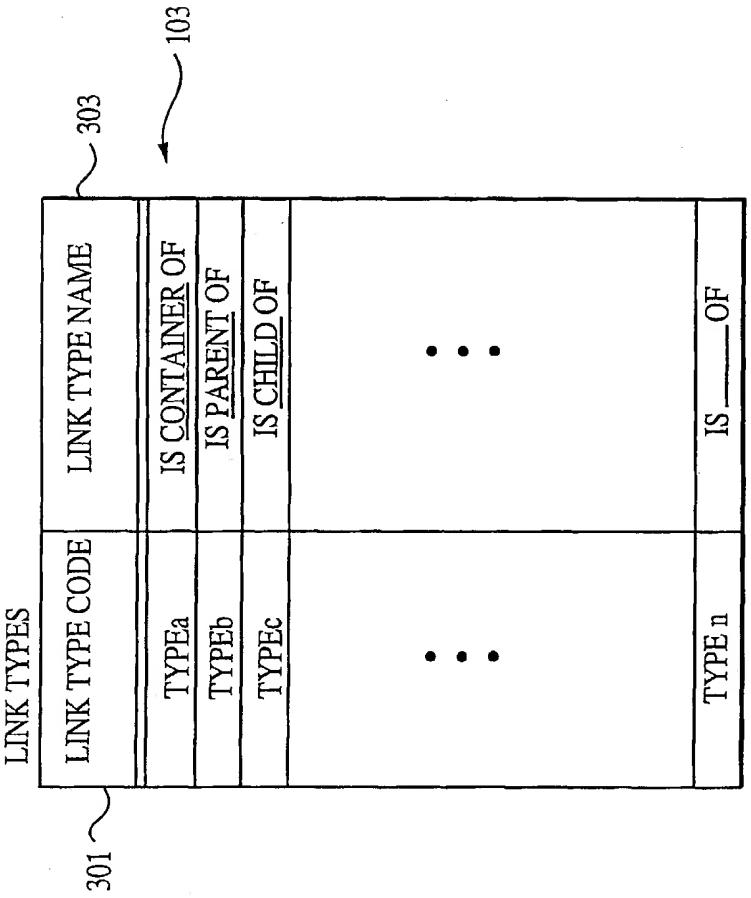

103
303
LINK TYPES
LINK TYPE CODE
LINK TYPE NAME
TYPEa
IS CONTAINER OF
TYPEb
IS PARENT OF
TYPEc
IS CHILD OF
TYPE n
IS ____OF
301

FIG. 3

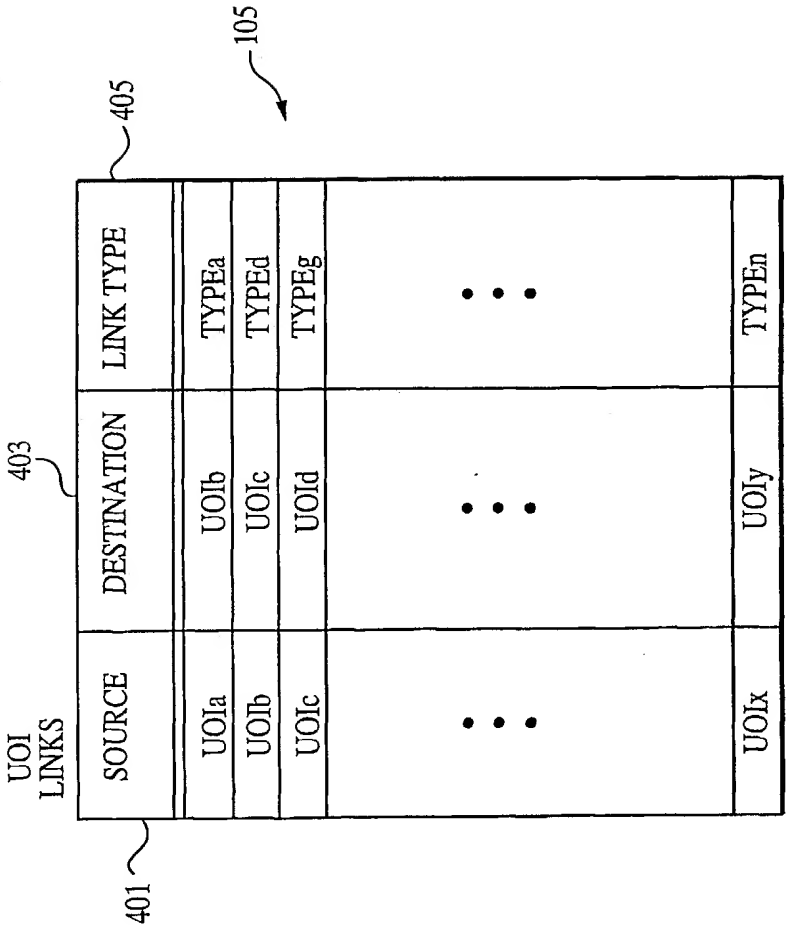

105
405
403
401
UOI LINKS
SOURCE
DESTINATION
LINK TYPE
UOIa
UOIb
UOIc
UOIb
UOIc
UOId
TYPEa
TYPEd
TYPEg
UOIx
UOIy
TYPEn

FIG. 4

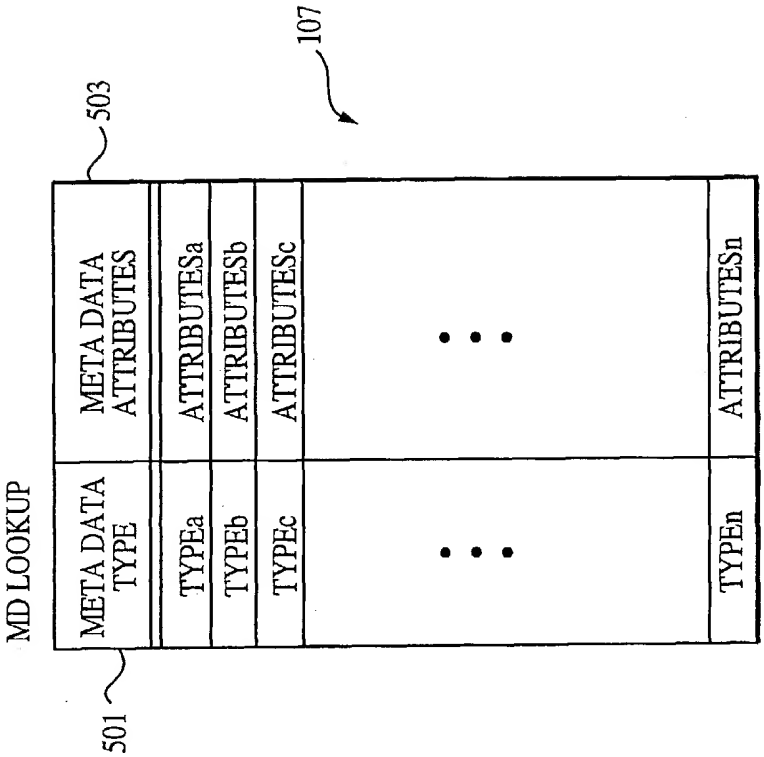

MD LOOKUP
META DATA TYPE
META DATA ATTRIBUTES
TYPEa
TYPEb
TYPEc
TYPEn
ATTRIBUTESa
ATTRIBUTESb
ATTRIBUTESc
ATTRIBUTESn
501
503
107

FIG. 5

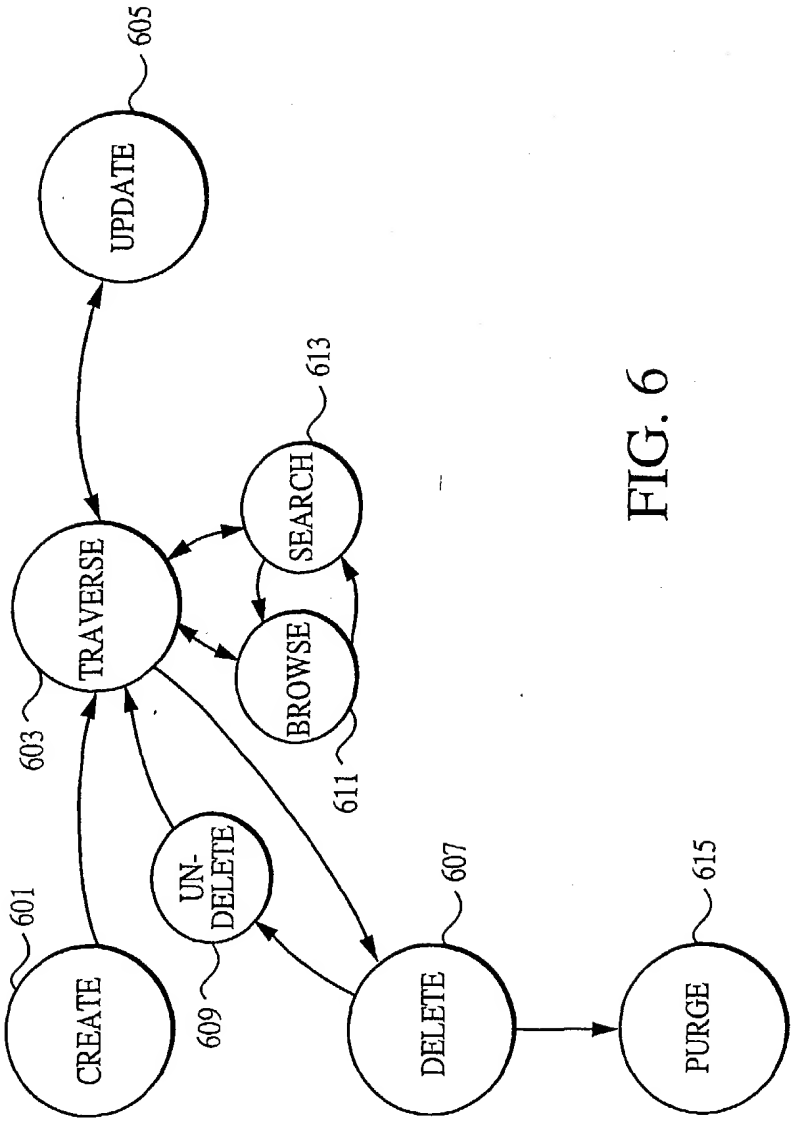

605
UPDATE
613
SEARCH
TRAVERSE
603
BROWSE
611
601
CREATE
UN-
DELETE
609
607
DELETE
615
PURGE

FIG. 6

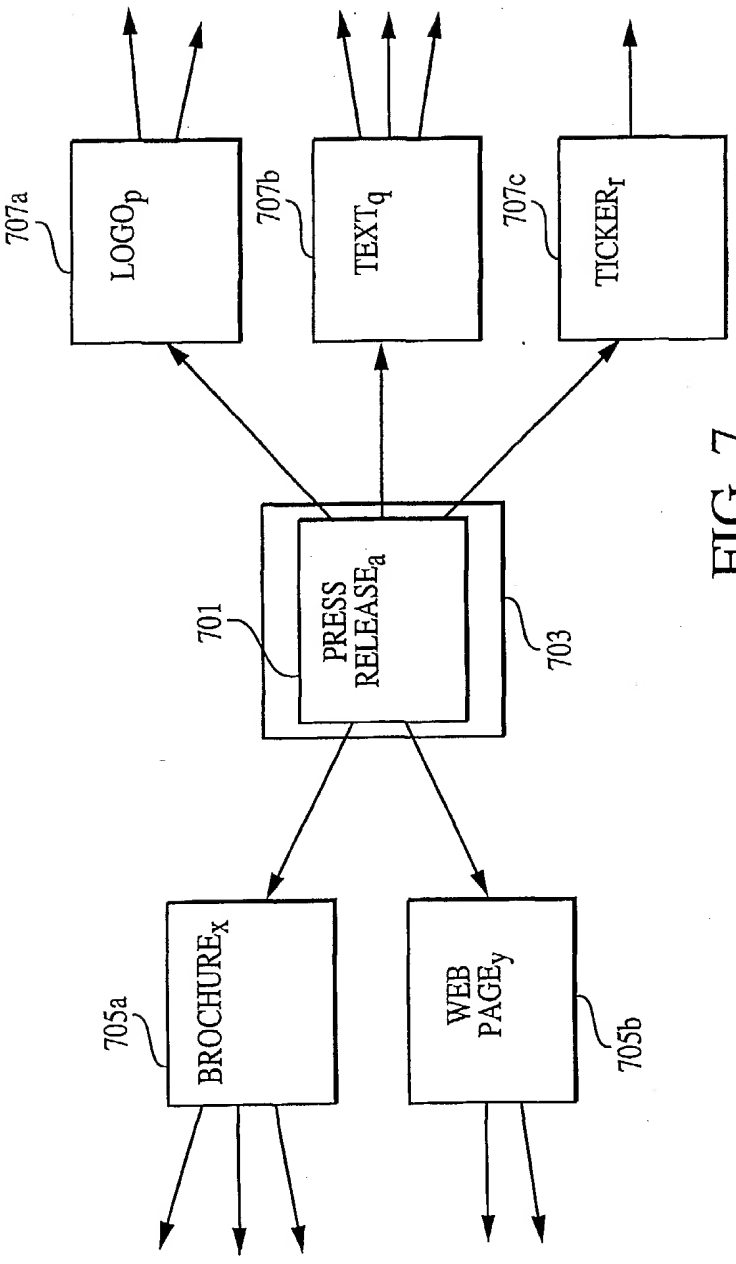

LOGO$_p$
TEXT$_q$
TICKER$_r$
707a
707b
707c
PRESS RELEASE$_a$
701
703
BROCHURE$_x$
WEB PAGE$_y$
705a
705b

FIG. 7

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US01/18784

A. CLASSIFICATION OF SUBJECT MATTER

IPC(7) : G06F 17/00
US CL : 707/100

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
U.S. : 707/100, 200

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
EAST

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 5,303,379 A (KHOYI et al) 12 April 1994 (12.04.1994), ALL.	1-20
Y	US 5,721,911 A (HA et al) 24 February 1998 (24.02.1998), ALL.	1-20

☐ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
05 September 2001 (05.09.2001)	27 SEP 2001
Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231 Facsimile No. (703)305-3230	Authorized officer Uyen T Le Peggy Harrod Telephone No. 703-305-3900

Form PCT/ISA/210 (second sheet) (July 1998)